Received January 30, 2019, accepted February 23, 2019, date of publication March 8, 2019, date of current version March 25, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2902724

Sheep Identification Using a Hybrid Deep Learning and Bayesian Optimization Approach

AYA SALAMA, ABOUL ELLAH HASSANIEN, AND ALY FAHMY
Faculty of Computer Science and Information Technology, Cairo University, Cairo 12613, Egypt

Corresponding author: Aya Salama (aya.salama@aucegypt.edu)

ABSTRACT Sheep are considered a necessary source of food production worldwide. Therefore, the sheep identification is vital for managing breeding and disease. Moreover, it is the only guarantee of an individual's ownership. Therefore, in this paper, sheep identities were recognized by a deep convolutional neural network using facial bio-metrics. To obtain the best possible accuracy, different neural networks designs were surveyed and tested in this paper. The Bayesian optimization was used to automatically set the parameters for a convolutional neural network; in addition, the AlexNet configuration was also examined in this paper. In this paper, the sheep recognition algorithms were tested on a data set of 52 sheep. Not more than 10 images were taken of each sheep in different postures. Thus, the data augmentation methodologies such as rotation, reflection, scaling, blurring, and brightness modification were applied; 1000 images of each sheep were obtained for training and validation. The experiments conducted in this paper achieved an accuracy of 98%. Our approach outperforms previous approaches for sheep identification.

INDEX TERMS Bayesian optimization, convolutional neural network, deep learning.

I. INTRODUCTION

Automatic sheep identification is now considered a necessity, but has been poorly studied by researchers. Ear tags are the traditional method usually used for sheep identification. However, this method has proved inefficient. Ear tags can be either lost or their numbers can be obscured due to the environments in which sheep live [1]. Most farmers and sellers depend on intuition to estimate sheep identity, which is prone to mistakes. It is important to accurately identify sheep during sheep selection for breeding and stock management. Moreover, it is important to track individuals with disease for treatment and for disease management, especially if there is an epidemic disease. In addition, buyers sometimes keep their sheep on the farm for some time; as a result, they have no guarantee of which animal they have bought. For this reason, to provide buyers, sellers and farmers an efficient way to recognize each individual in a large group of sheep, an automatic real-time sheep identification approach is proposed in this paper. Sheep facial biometrics include many significant features that can be used for identification such as muscles, the eyes, mouth and many hidden features [2]. Therefore, facial biometrics are very promising and efficient features for sheep recognition. As a result, the approach proposed in this paper was based on sheep facial images.

The associate editor coordinating the review of this manuscript and approving it for publication was Oguz Elibol.

Deep learning is a hot topic in the field of machine learning and pattern recognition, whose efficiency has been proven in multiple tasks. Deep learning is nothing but a machine learning methodology that learns by example to classify images, text or even sound. Deep learning can outperform humans in most of its applied tasks [3]. Moreover, convolutional neural networks (CNNs) are considered the most popular effective deep learning approach in all previous scientific literature [4]–[6]. Accordingly, CNNs have been used in this work to classify sheep. However, design of CNNs is a bottleneck that is frequently challenging in using CNNs. It is complicated to design a CNN to suit a specific classification task [7]. For this reason, this paper focused on finding the appropriate design for CNNs by experimenting with existing architectures and studying optimization techniques.

Particle swarm optimization (PSO) is similar to genetic algorithms because it initiates random solutions that are then adjusted until the optimal solution with the best fitness is reached. PSO basically utilizes a group of particles, or swarms that mimics a swarm's behavior to find the best solution by having its members interact with each other. The particles or swarms of the optimization model iterate, while seeking the optimal solution. The swarm is represented by a vector that changes iteratively, saving the best solutions and changing their priorities [8].

The gravitational search algorithm (GSA) is another optimization algorithm that updates a collection of possible solutions through all possible generations. These solutions are initially distributed randomly. However, in this optimization algorithm, solutions that have a relatively higher fitness function attract each other similar to gravity until the optimal solution is reached [9], [10]. PSO and GSA are two algorithms that can be heterogeneously hybridized to combine their functionalities. Mirjalili *et al.* [11] were able to implement these algorithms in parallel. The main goal of this hybrid approach is to prevent the method from becoming stuck in local optimum in contrast to backpropagation [12]. However, backpropagation is preferable for training deep neural networks because backpropagation approximates the partial derivatives of the error. A literature survey found that both PSO and GSA are proven optimization techniques that can be used when training neural networks. However, when substituted for backpropagation, these approaches require long processing times and large processing capabilities. In [8], PSO was used to optimize a CNN's parameters instead of to replace backpropagation. However, heavy computational loads are still required. In addition, the final classification results are unsatisfactory. The maximum accuracy reached in image classification is 80.15%. Moreover, the goal of this paper is to produce an application that can be released on mobile phones to be used by all types of users.

Thus, Bayesian optimization was adopted in this study to help in searching for the best CNN architecture. Bayesian optimization is more efficient than PSO because it requires fewer trials and fewer initial parameters [13].

In [2], sheep were recognized using a cosine distance classifier trained on facial images of 50 sheep whose ages ranged from 3 to 4 years. The cosine classifier depends on the cosine distance threshold for the classification task. Each sheep was represented by 7 images taken at a forward-facing posture with a black background. The faces of the sheep were cleaned of dirt and all possible sources of noise before being imaged. Sheep were also held using special tools so that a certain fraction of each sheep face was within the image. Therefore, this approach required considerable human intervention for image acquisition. A Canon professional PowerShot camera with resolution of 1024 × 768 pixels was used. The highest accuracy achieved was 96%.

The rest of this paper is organized as follows: Section 2 describes the camera settings. Section 3 explains the theoretical background behind most of the methodologies adopted in this paper. Section 4 then provides the details of the proposed approach for sheep identification. The experimental results and evaluation are elaborated upon in section 5. Finally, section 6 presents conclusions and some directions for future work.

II. CAMERA SETTINGS

Fifty-two sheep were imaged by an ordinary mobile camera. The sheep ranged in age from 5 months to 5 years and had different physiologies. All the sheep were of the Barqi breed, which originated in Libya. Facial images were taken in

FIGURE 1. Sheep samples.

different postures, as shown in Figure 1, and at slightly varying distances from the camera, with a 1.5 meters average distance. The camera settings are illustrated in Figure 2. Five to ten face images were taken of each sheep in different postures. However, for each of the 52 sheep, a total of 1000 images were produced by means of data augmentation.

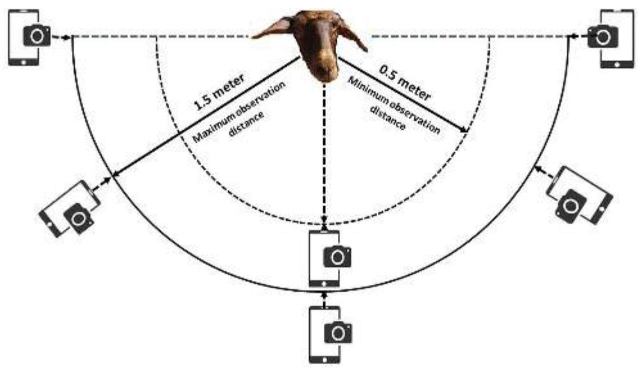


FIGURE 2. Camera model settings.

III. THEORY AND BACKGROUND

A. DEEP LEARNING

Deep learning is a well-known machine learning approach that can perform direct classification from images [14]. However, unlike other machine learning approaches, deep learning is based on abstraction by applying hierarchical approaches with deep layers. Accordingly, it is called deep. The performances of deep learning techniques exceed those of all other techniques on any image recognition task. Although deep learning requires numerous training images, it extracts its own features without the need for any supervision. As a result, deep learning yields more accurate results. Deep learning trains a hierarchy through a sufficient number of iterations, propagating from given input to output until it reaches adequate accuracy. For this reason, it is also called deep neural learning as it simulates the neurons in brains. Therefore, almost all deep learning techniques involve neural network models.

B. CONVOLUTIONAL NEURAL NETWORKS

CNNs are the most powerful and effective deep learning technique. CNNs are a form of artificial neural networks (ANNs). Artificial neural networks basically consist of an input layer, hidden layers and an output layer, as shown in Figure 3. All the layers consist of artificial neurons that mimic brain neurons. Each neuron has attached weights so that data are transferred from the input layer to the output layer through the hidden layers [15], [16]. These weights are iteratively adjusted by means of an activation function that takes the sum of the input weights as input. Networks iterate to minimize the error. Adding additional hidden layers creates what is called a deep neural network. However, CNNs can take entire images as input; unlike a typical ANN, CNNs scale well (Figure 4).

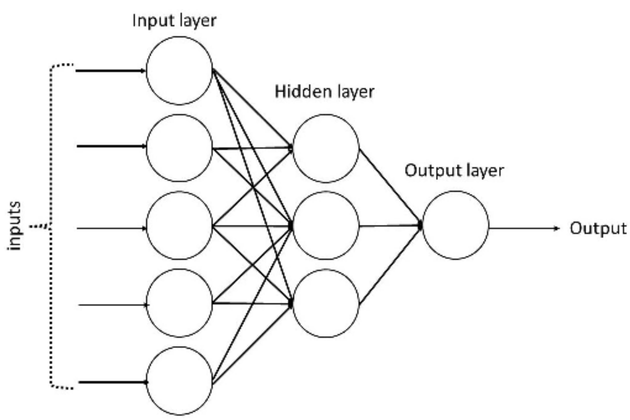


FIGURE 3. Neural network.

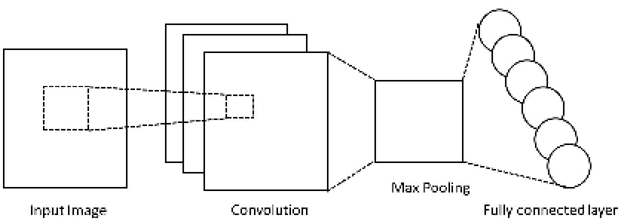


FIGURE 4. Convolutional neural network.

The architecture of CNNs mainly consists of an input layer, a convolutional Layer, a rectified linear units (ReLU) layer, a pooling Layer, and a fully-connected layer. The input layer usually takes an input image of (height × width × number of channels). For example, an RGB image has three color channels. Convolutional layers are the core layers of a CNN because they contain convolutional filters, which are also called kernels. Each of these filters convolves the entire image to produce an activation function that responds to certain features, such as edges and colors. Then, a ReLU layer is usually adopted because it applies the ReLU activation function to speed up the training process. The pooling layer progressively down-samples the input image to prevent overfitting because it removes redundant information. Finally, the fully connected layers come after multiple convolutional and pooling layers. In these layers, all the neurons are connected to all the activation functions of the previous layer to recognize large patterns. The final layer determines the corresponding class by feature combination. However, its architecture may differ depending on application or data. Therefore, network can have one or two convolutional layers or it can be complicated network with hundreds of convolutional and fully connected layers.

C. BAYESIAN OPTIMIZATION TECHNIQUE

Bayesian optimization is an efficient technique in the machine learning field [17] and the best choice for an expensive objective function [18]. Bayesian optimization is a black-box technique that aims to minimize or maximize any given objective function by constructing a probabilistic model. Bayesian optimization is composed of a probabilistic model and a loss function. It aims to model the objective function f to specify its distribution.

$$x_{new} = arg_x \in X^{max/min} f(x) \tag{1}$$

where X is any given design space of interest. This model is used for making efficient sampling decisions, and is sequentially updated. Furthermore, it applies an acquisition function to maintain both exploration and exploitation capabilities. The acquisition function selects the appropriate candidates for the next selection. A Gaussian process is often selected to acquire some of the parameters needed for the objective function. The loss function shows the optimality of the running sequence. The Bayesian optimization process is summarized by the pseudo code in Figure 5.

Bayesian Optimization Algorithm

Assuming goal is to maximize unknown function f(x) on data D:

for n loops do
-> select new x_{n+1} by optimization of α which is an acquisition function x_{n+1} = max α(x; D_n)
-> get new observation y_{n+1} from objective function
-> augment data $D_{n+1} = \{D_n, (x_{n+1}, y_{n+1})\}$
-> update model
end for

FIGURE 5. Bayesian optimization algorithm.

D. ALEXNET

AlexNet is a pre-trained state-of-the-art CNN designed by Alex Krizhevesky. It has been used in experiments on different benchmarks in various fields and it preceded all other deep learning approaches. For this reason, AlexNet architecture has been used in many image classification experiments [19], [20]. AlexNet is composed of 5 convolutional layers and 2 fully connected layers. The input image size is 227 × 227. The first convolutional layer operates with

96 different 11 × 11 filters while the max pooling layer operates with 3 × 3 filters. The second layer has 5 × 5 filters. The third, fourth and fifth layers have 3 × 3 filters.

IV. PROPOSED APPROACH

The proposed approach is summarized in Figure 6 and explained in detail in the following subsections.

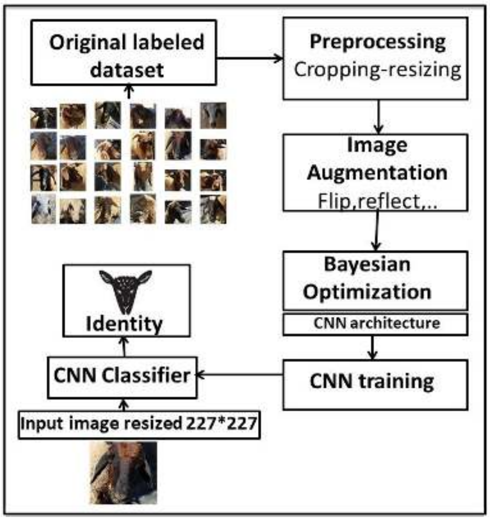


FIGURE 6. Proposed approach model.

A. PRE-PROCESSING

As mentioned above, facial images of various sizes were taken for 52 sheep at different postures. Therefore, the images had to be resized to match the input size compatible with the neural network. In the proposed approach, the fixed input size is 227 × 227. Moreover, most of the images contained noisy features that can hinder efficient classification. For this reason, the collected sheep images were cropped to display only on the face of the sheep, as shown in Figure 7. In addition to resizing and cropping images, some of the tested approaches required grey-scale image inputs. Accordingly, the RGB images were also transformed into grey-scale images.

FIGURE 7. Image samples after pre-processing.

B. AUGMENTATION

Deep learning approaches require a sufficient number of training images to boost their performance [20]. For this reason, all previous approaches that used CNNs to solve visual recognition problems using few training images had to resort to image augmentation in which new images are created from the existing training images by augmentation techniques and used to supplement the original training images. The augmentation techniques used in the proposed approach in this paper are rotation, reflection, scaling, blurring and brightness adjustment. Images were flipped horizontally and vertically. For rotation, images were rotated randomly to different degrees. Scaling was also applied to images in either the x direction or the y direction. Images were adjusted to different degrees of brightness and blurred by a Gaussian variance function whose values ranged from 0.1 to 0.9. Moreover, through image augmentation, CNNs are less prone to memorize the training images, thus avoiding overfitting. Samples of the augmented images are presented in Figure 8.

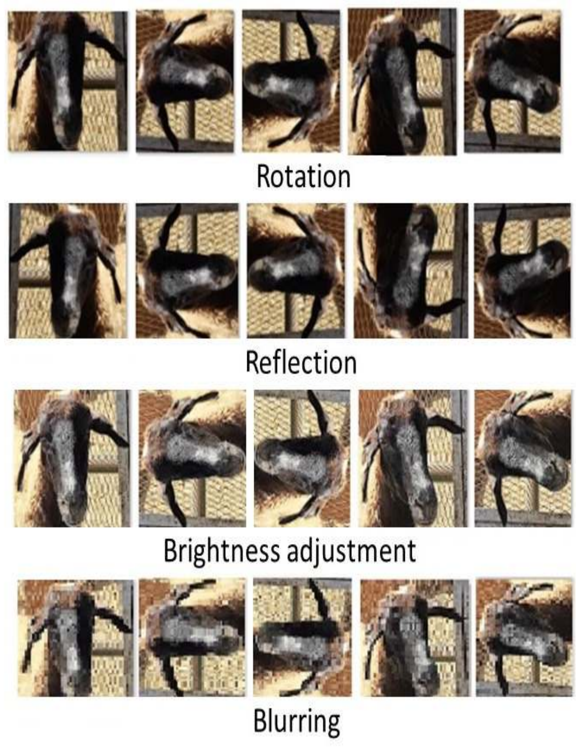


FIGURE 8. Augmented image samples.

C. BAYESIAN OPTIMIZATION TO AUTOMATICALLY DESIGN CNNS

Bayesian optimization techniques have been proven to excel in the field of machine learning. Correspondingly, Bayesian optimization was tested in our proposed approach for automatic parameter selection. The parameters to be optimized were the number of convolutional layers, the initial learning rate, momentum, and regularization strength. The objective function used the training set to train the CNN and used the validation set to test the accuracy of the classifier to prevent overfitting. The dataset of 52000 images was split so that 80% of the images were used as training set and 20% were used as the testing set. The validation set was 5000 images of the selected training set. The objective function was used when training the CNN to obtain the classification error of the tested architecture for each iteration using the validation set. Therefore, the goal was to find the most optimal solution by minimizing the classification error. The objective function evaluation was repeated until the smallest error was reached as shown in Figure 9.

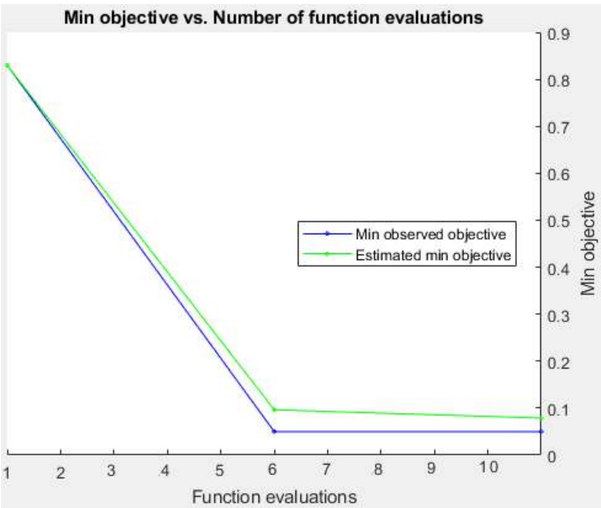


FIGURE 9. Objective function evaluations.

D. CNN

The optimal CNN architecture that gave the best accuracy was then chosen based on the output of the Bayesian optimization techniques. The chosen network architecture was then loaded and tested using the testing set. The convolutional layers were always padded to correspond to the input through epochs. Max pooling layers were down-sampled at a factor of 2 using kernels, and the number of filters was chosen to be proportional to the number of convolutional layers. During training, the network was validated after each epoch. After each epoch, the images were augmented to keep the network from memorizing an image's features and prevent overfitting. The training cycle was composed of 40 epochs, and each epoch had 36 iterations. To reduce the noise during parameter updating, the learning rate was reduced by a factor of 10 across epochs. The final CNN was saved to be used as a classifier for the unseen sheep images.

V. EXPERIMENTAL RESULTS AND EVALUATION

The final CNN was generally outstanding at the image classification task. However, developing a CNN structure is never easy task because time is required to try different structures until the best classification rate is reached. After so many trials, the best simple deep CNN achieved an accuracy of 48.87% accuracy. The accuracy of classification methods is determined as follows:

$$C_{acc} = \frac{CP}{TP} * 100 \tag{2}$$

where C_{acc} is the classification accuracy, CP is the number of correctly predicted images, and TP is the total number of predictions.

The CNN was composed of one input layer of size $227 \times 227 \times 3$, and the data were shuffled at the beginning of each epoch. Three convolutional layers with 3×3 filters were used. Each convolutional layer was followed by a ReLU layer and a max pooling layer. After these layers, 10 full layers had been used, resulting in the last layer having 52 outputs corresponding to the 52 classes of the sheep in the collected data. Initial learning rate was 0.01, maximum number of epochs was 4 and stochastic gradient descent with momentum (SGDM) was used. Momentum was 0.9000 and Regularization was 1.0000e-04. However, 48.87% is dramatically below any acceptable accuracy. Figure 10 shows how the accuracy changes with the number of iterations. For all the previously mentioned reasons, using an optimization technique was required.

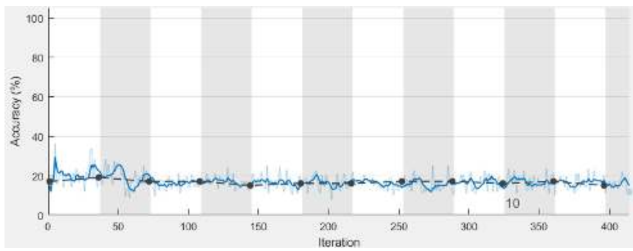


FIGURE 10. Accuracy achieved by the simple CNN versus the number of iterations.

In this paper, two different methodologies were analyzed to generate the CNN structure with the best accuracy; these CNNs were tested on the testing set. The first tested methodology generated CNN parameters using Bayesian optimization. The second methodology involved training AlexNet on a sheep dataset and comparing its results to the results obtained from the approach proposed in this work. A Bayesian optimization function was used to determine the CNN structure. The parameters to optimize were the number of convolutional layers, initial learning rate, momentum, and regularization strength. Evaluations of the objective function were repeated until the best structure was obtained. Some of the acceptable evaluations that gave low classification errors are shown in Table 1.

TABLE 1. Optimization parameters through evaluations.

Hidden Layers	Learning Rate	Momentum	Regularization
2	0.0021992	0.86792	6.0813e-08
6	0.0203	0.91225	8.8774e-08
6	0.0010172	0.83681	0.0045877

The best estimated structure found by Bayesian optimization is illustrated in Table 2.

TABLE 2. Optimal parameters.

Hidden Layers	Learning Rate	Momentum	Regularization
6	0.0010172	0.83681	0.0045877

The CNN with the structure generated from Bayesian optimization was then loaded and tested with the testing set; this model achieved an accuracy of 98%. Figure 11 shows its evolution over all the iterations, while Figure 12 shows the evolution of the loss function towards its minimum over these iterations. This trained CNN was then saved to classify any new image for sheep identification. The previous approach to sheep identification, which used a cosine classifier, achieved an accuracy of only 96% [2]. In addition, special cameras and special tools are needed to obtain specific sheep postures. Therefore, the proposed approach outperforms the only other sheep identification approach present in the literature.

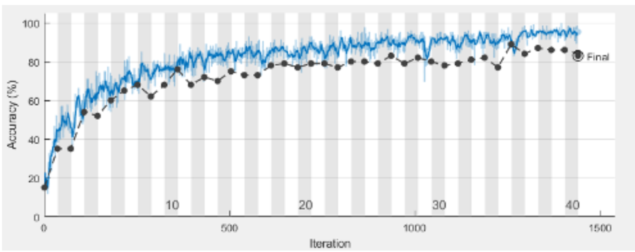


FIGURE 11. Accuracy achieved by CNN designed by Bayesian optimization through iterations.

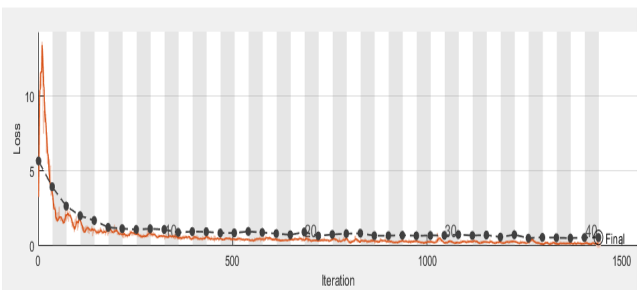


FIGURE 12. Parameters updated by Bayesian optimization settling down closer to a minimum of the loss function through epochs.

As mentioned previously, AleXNet settings were used heavily in most image classification tasks as they have proved their success in training on any new data set. For this reason, AlexNet was also trained on the sheep dataset used in this work. The images used were already resized and augmented, so they were ready to be input into the AlexNet CNN. This AlexNet model had 23 layers and the number of fully connected layers was 52 to support the number of classes. Stochastic gradient descent with momentum was used as an optimizer with global learning rate of 0.001. Moreover, Maximum number of epochs was 20 for fine-tuning. Furthermore, the parameters were updated using a subset of data of size 64. The dataset of 52000 images was split in the same way as in the proposed approach, i.e., 80% of the images were used as a training set, and 20% were used as a validation set. The accuracy achieved by AlexNet was 97.5%, which is less than the accuracy achieved using the Bayesian optimization; however, these results are very close. Accordingly, AlexNet could also be used to identify sheep. The accuracies of the different tested approaches to sheep identification are summarized in Table 3. All the experiments were conducted on Matlab R2018a with the environment windows 10, Intel Core i7 and 8G memory.

TABLE 3. Comparison of accuracy results.

Proposed Approach	CNN without optimization	Cosine	AlexNet
98%	48.87%	96%	97.5%

VI. CONCLUSION AND FUTURE WORK

Deep learning is a hot topic, and CNNs are the most efficient deep learning approach for visual recognition problems. However, designing an efficient CNN is a complicated task. For this reason, the use of optimization techniques is now essential for setting CNN parameters. Bayesian optimization was used in the proposed approach to set the CNN parameters and design its structure for use in recognizing individual sheep. The accuracy achieved by the hybrid approach of a CNN and Bayesian optimization was 98%; thus, the proposed approach that used cosine distance classifier for sheep identification, which achieved an accuracy of only 96%. Moreover, the previous approach required special tools and special environments. AlexNet is a pre-trained CNN that has demonstrated its ability to be effectively trained on images in almost all domains and to provide highly accurate results. Therefore, we also tested the AlexNet architecture on the sheep data set, and it achieved an accuracy of 97.5%, which is slightly less than that achieved by the proposed approach. For future work, the proposed approach will be combined with weight and age estimation function and released as a mobile application. This application will be able to provide all possible information about sheep by using image recognition. In addition, images will be collected for sheep breeds other than Barqi sheep, and the CNN will be trained to recognize a sheep's breed before its identity.

REFERENCES

[1] B. T. Koik and H. Ibrahim, "A literature survey on animal detection methods in digital images," *Int. J. Future Comput. Commun.*, vol. 1, no. 1, p. 24, Jun. 2012.

[2] G. P. Corkery, U. A. Gonzales-Barron, F. Butler, K. M. Donnell, and S. Ward, "A preliminary investigation on face recognition as a biometric identifier of sheep," *Trans. ASABE*, vol. 50, no. 1, pp. 313–320, 2007.

[3] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, May 2015.

[4] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

[5] L. Chen, X. Guo, and C. Geng, "Human face recognition based on adaptive deep convolution neural network," in *Proc. 35th Chin. Control Conf.*, Chengdu, China, Jul. 2016, pp. 6967–6970.

[6] P. Priya and S. Jain, "A survey on face recognition using convolution neural networks," *Int. J. Softw. Hardw. Res. Eng.*, vol. 5, no. 9, Sep. 2017. [Online]. Available: https://ijournals.in/wp-content/uploads/2017/09/2.IJ-5911-PriyaTami.compressed.pdf

[7] B. Wang, Y. Sun, B. Xue, and M. Zhang, "Evolving deep convolutional neural networks by variable-length particle swarm optimization for image classification," in *Proc. IEEE Congr. Evol. Comput.*, Jul. 2018, pp. 1–8.

[8] T. Yamasaki, T. Honma, and K. Aizawa, "Efficient optimization of convolutional neural networks using particle swarm optimization," in *Proc. IEEE 3rd Int. Conf. Multimedia Big Data*, Apr. 2017, pp. 70–73.

[9] E. Rashedi, H. Nezamabadi-Pour, and S. Saryazdi, "A gravitational search algorithm," *Inf. Sci.*, vol. 179, no. 13, pp. 2232–2248, Jun. 2009.

[10] S. Yazdani, H. Nezamabadi-Pour, and S. Kamyab, "A gravitational search algorithm for multimodal optimization," *Swarm Evol. Comput.*, vol. 14, pp. 1–14, Feb. 2014.

[11] S. Mirjalili, S. Z. M. Hashim, and H. M. Sardroudi, "Training feedforward neural networks using hybrid particle swarm optimization and gravitational search algorithm," *Appl. Math. Comput.*, vol. 218, no. 22, pp. 11125–11137, Jul. 2012.

[12] S. Jiang, Z. Ji, and Y. Shen, "A novel hybrid particle swarm optimization and gravitational search algorithm for solving economic emission load dispatch problems with various practical constraints," *Int. J. Elect. Power Energy Syst.*, vol. 55, pp. 628–644, Feb. 2014.

[13] S. Snoek, H. Larochelle, and R. Adams, "Practical Bayesian optimization of machine learning algorithms," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 25, 2012, pp. 2951–2959.

[14] T.-H. Chan, K. Jia, S. Gao, J. Lu, Z. Zeng, and Y. Ma, "A simple deep learning baseline for image classification," *IEEE Trans. Image Process.*, vol. 24, no. 12, pp. 5017–5032, Dec. 2015.

[15] K. Simonyan and A. Zisserman. (Sep. 2014). "Very deep convolutional networks for large-scale image recognition." [Online]. Available: https://arxiv.org/abs/1409.1556

[16] J. Donahue *et al.*, "DeCAF: A deep convolutional activation feature for generic visual recognition," in *Proc. ICML*, Jan. 2014, pp. 647–655.

[17] B. Shahriari, K. Swersky, Z. Wang, R. Adams, and N. de Freitas, "Taking the human out of the loop: A review of Bayesian optimization," *Proc. IEEE*, vol. 104, no. 1, pp. 148–175, Jan. 2016.

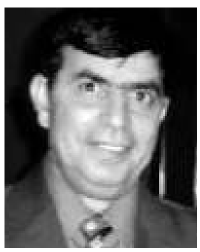
[18] Y. Zhang, K. Sohn, R. Villegas, G. Pan, and H. Lee, " Improving object detection with deep convolutional networks via Bayesian optimization and structured prediction," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 249–258.

[19] M. Z. Alom *et al.*. (Mar. 2018). "The history began from AlexNet: A comprehensive survey on deep learning approaches." [Online]. Available: https://arxiv.org/abs/1803.01164

[20] L. Perez and J. Wang. (Dec. 2017). "The effectiveness of data augmentation in image classification using deep learning." [Online]. Available: https://arxiv.org/abs/1712.04621

AYA SALAMA was born in Cairo, Egypt, in 1987. She received the B.S. degree in computer science from Ain Shams University, Cairo, in 2008, and the M.S. degree in computer science from the American University in Cairo, in 2014. She is currently pursuing the Ph.D. degree in computer science with Cairo University. From 2011 to 2017, she was a Research Assistant with the American University in Cairo. Her research interests include machine learning, pattern recognition, computer vision, and image processing. Her awards and honors include University Fellowship in the Masters of Science program in computer science with the American University in Cairo, a Certificate of Academic Honor from the American University in Cairo, and the Best Presentation Award at the International Conference on Machine Vision (London).

ABOUL ELLAH HASSANIEN received the B.Sc. degree (Hons.) and the M.Sc. degree from the Pure Mathematics and Computer Science Department, Faculty of Science, Ain Shams University, Cairo, Egypt, in 1986 and 1993, respectively, and the Ph.D. degree from the Department of Computer Science, Graduate School of Science Engineering, Tokyo Institute of Technology, Japan, in 1998. He is currently a Professor with the IT Department, Faculty of Computers and Information, Cairo University, and the Chair of the Technology Center of Blind and Visual Impaired People. He has authored/co-authored over 160 research publications in peer-reviewed reputed journals, book chapters, and conference proceedings. His research interests include rough set theory, wavelet theory, computational intelligences, medical image analysis, information security, intelligent environment, and multimedia processing. He has served as a Program Committee Member of various international conferences, and as a Reviewer for various international journals. He has received the Excellence Younger Researcher Award from Kuwait University, for the academic year (2003–2004). He has Guest Edited many special issues for international scientific journals. He has directed many funded research projects. He is the Editor and Co-Editor of over 20 books in the area of rough computing, computational intelligence, social networks, bioinformatics, and E-commerce.



ALY FAHMY received the B.Sc. degree (Excellent with Honor Grade) in computer engineering from the Computer Department, Military Technical College (M.T.C), in 1972, the DPL - Diploma: General Purpose Simulation from the Computer Department, M.T.C, in 1973, the M.Sc. degree in logical database systems from the Computer Department, E.N.S.A.E, Toulouse, France, in 1976, and the Ph.D. degree in artificial intelligence control of automatic deductions for logic-based systems from the Computer Department, Centre of Research and Studies, C.E.R.T, Toulouse, France, in 1979. He is currently a Professor with the Faculty of Computers and Information, Cairo University.